

REMARKS

In the Official Action mailed **January 5, 2005**, the Examiner reviewed claims 1, 3-5, 7-11, 13-15, 17-21, 23-25, and 27-33. Claims 1, 3-5, 7-11, 13-15, 17-21, 23-25, and 27-33 were rejected under 35 U.S.C. §103(a) as being unpatentable over Maier et al. (USPN 5,625,815, hereinafter “Maier”) in view of Elmasri et al. (*Fundamentals of Database System*, Copyright 2000, hereinafter “Elmasri”)..

Rejections under 35 U.S.C. §103(a)

Independent claims 1, 11, and 21 were rejected as being unpatentable over Maier in view of Elmasri.

Applicant respectfully points out that Maier teaches away from the present invention. Note that there are essentially two types of database transactions: those that **change** the database (e.g., CREATE, ADD, DELETE), and those that **do not change** the database (e.g., SELECT).

Maier is specifically directed towards auditing transactions that **change** the database. Specifically, Maier discloses that a “*transaction manager generates and stores an audit trail, each audit entry denoting a database table or index record event, such as an **addition, deletion or alteration** of a specified database table or index record*” (see Maier, col. 2, lines 34-38). Furthermore, the system of Maier uses the audit record to “*apply the **change** noted in the audit record to the NewPartition*” (see FIG. 4A, 222, FIG. 4B, 224, FIG. 5A, 232, FIG. 5B, 234, FIG. 6A, 242, FIG. 6B, 244, FIG. 7B, 253).

In contrast, the present invention can selectively audit read-only transactions that **do not change** the database. Specifically, the present invention creates “*an audit record for rows in relational tables that are accessed by the **query**, and that satisfy an auditing condition*” (see page 3, lines 13-15). Note that, a database query (e.g., SELECT) can be a read-only transaction that **does not**

change the database. For example, see Garcia-Molina et al. (“Read-only transactions in a distributed database,” vol. 7, Issue 2, pp. 209-234, ACM Transactions on Database Systems, June 1982, hereinafter “Garcia-Molina”). Specifically, the present invention selectively audits read-only transactions because, *“a database system generally maintains a log of changes to database tables, so providing an auditing mechanism for these changes in addition to the log may be unnecessary”* (see page 9, lines 11-13).

Note that, for database security reasons, it is absolutely critical to selectively audit database transactions (e.g., queries) that access sensitive data, even if they are read-only transactions that do not change the database.

Moreover, the process of selectively auditing read-only transactions (e.g., SELECT) is not obvious. This is because database audits are typically used to keep track of database changes for maintenance and integrity purposes. In contrast, the present invention selectively audits read-only transactions to improve database security. Furthermore, the present invention involves the complex operations of modifying the *“query to insert monitoring logic”* (see FIG. 3, 304), *“view merging through optimization layer”* (see FIG. 3, 305), and creating an *“audit record for rows that are accessed by the query and that satisfy the auditing condition”* (see FIG. 3, 310).

Accordingly, Applicant has amended independent claims 1, 11, and 21 to further clarify that the query can be a read-only transaction. Note that it is well known in the art that a query can be a read-only transaction that does not change any tables in the database. Specifically, these amendments find support in Garcia-Molina.

Hence, Applicant respectfully submits that independent claims 1, 11, and 21 as presently amended are in condition for allowance. Applicant also submits that claims 3-5, 7-10 and 31, which depend upon claim 1, claims 13-15, claims 17-20 and 32, which depend upon claim 11, and claims 23-25, 27-30 and 33, which depend upon claim 21, are for the same reasons in condition for allowance and for reasons of the unique combinations recited in such claims.

CONCLUSION

Applicant respectfully requests that a timely Notice of Allowance be issued in this case.

Respectfully submitted,

By



A. Richard Park

Registration No. 41,241

Date: February 16, 2005

A. Richard Park
PARK, VAUGHAN & FLEMING LLP
2820 Fifth Street
Davis, CA 95616
Tel: (530) 759-1661
FAX: (530) 759-1665


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

Search: [The ACM Digital Library](#) [The Guide](#)

THE GUIDE TO COMPUTING LITERATURE

 [Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Read-only transactions in a distributed database

Full text  [Pdf \(2.16 MB\)](#)

Source **ACM Transactions on Database Systems (TODS)** [archive](#)

Volume 7 , Issue 2 (June 1982) [table of contents](#)

Pages: 209 - 234

Year of Publication: 1982

ISSN:0362-5915

Authors [Hector Garcia-Molina](#) Princeton Univ., Princeton, NJ
[Gio Wiederhold](#) Stanford Univ., Stanford, CA

Publisher ACM Press New York, NY, USA

Additional Information: [abstract](#) [references](#) [citations](#) [index terms](#) [collaborative colleagues](#) [peer to peer](#)

Tools and Actions: [Discussions](#) [Find similar Articles](#) [Review this Article](#)
[Save this Article to a Binder](#) Display Formats: [BibTex](#) [EndNote](#)

DOI Bookmark: Use this link to bookmark this Article: <http://doi.acm.org/10.1145/319702.319704>
[What is a DOI?](#)

↑ ABSTRACT

A read-only transaction or query is a transaction which does not modify any data. Read-only transactions could be processed with general transaction processing algorithms, but in many cases it is more efficient to process read-only transactions with special algorithms which take advantage of the knowledge that the transaction only reads. This paper defines the various consistency and currency requirements that read-only transactions may have. The processing of the different classes of read-only transactions in a distributed database is discussed. The concept of R insularity is introduced to characterize both the read-only and update algorithms. Several simple update and read-only transaction processing algorithms are presented to illustrate how the query requirements and the update algorithms affect the read-only transaction processing algorithms.

↑ REFERENCES

Note: OCR errors may be found in this Reference List extracted from the full text article. ACM has opted to expose the complete List rather than only correct and linked references.

1 [Peter A. Alsberg , John D. Day, A principle for resilient sharing of distributed resources, Proceedings of the 2nd international conference on Software engineering, p.562-570, October 13-15, 1976, San Francisco, California, United States](#)

2 BERNSTEIN, P.A., ROTHNIE, J.B., GOOVMAN, N., AND PAPADIMITRIOU, C.A. The concurrency control mechanism of SDD-I: A system for distributed databases (the fully redundant case). IEEE Trans. Softw. Eng. 4, 3 (1978), 154-168.